



Elastic Images: Perceiving Local Elasticity of Images Through a Novel Pseudo-Haptic Deformation Effect

Ferran Argelaguet Sanz, David Antonio Gómez Jáuregui, Maud Marchal, Anatole Lécuyer

► To cite this version:

Ferran Argelaguet Sanz, David Antonio Gómez Jáuregui, Maud Marchal, Anatole Lécuyer. Elastic Images: Perceiving Local Elasticity of Images Through a Novel Pseudo-Haptic Deformation Effect. ACM Transactions on Applied Perception, 2013, 10 (3), pp.17:1–17:14. hal-00907775

HAL Id: hal-00907775

<https://hal.science/hal-00907775>

Submitted on 22 Nov 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Elastic Images: Perceiving Local Elasticity of Images through a Novel Pseudo-Haptic Deformation Effect

Ferran Argelaguet, Inria Rennes
David Antonio Gómez Jáuregui, Inria Rennes
Maud Marchal, Inria/INSA Rennes
Anatole Lécuyer, Inria Rennes

We introduce the Elastic Images, a novel pseudo-haptic feedback technique which enables the perception of the local elasticity of images without the need of any haptic device. The proposed approach focus on whether visual feedback is able to induce a sensation of stiffness when the user interacts with an image using a standard mouse. The user, when clicking on a Elastic Image, is able to deform it locally according to its elastic properties. To reinforce the effect, we also propose the generation of procedural shadows and creases to simulate the compressibility of the image and several mouse cursors replacements to enhance pressure and stiffness perception. A psychophysical experiment was conducted to quantify this novel pseudo-haptic perception and determine its perceptual threshold (or its Just Noticeable Difference). The results showed that users were able to recognize up to eight different stiffness values with our proposed method and confirmed that it provides a perceivable and exploitable sensation of elasticity. The potential applications of the proposed approach range from pressure sensing in product catalogs and games, or its usage in graphical user interfaces for increasing the expressiveness of widgets.

Categories and Subject Descriptors: H.5.2 [Information Interfaces and Presentation]: User Interfaces—*Evaluation/methodology*; H.5.2 [Information Interfaces and Presentation]: User Interfaces—*Haptic I/O*; I.3.6 [Computer Graphics]: Methodology and Techniques—*Interaction techniques*

General Terms: Human Factors

Additional Key Words and Phrases: Pseudo-Haptic, Texture, Elasticity, Stiffness

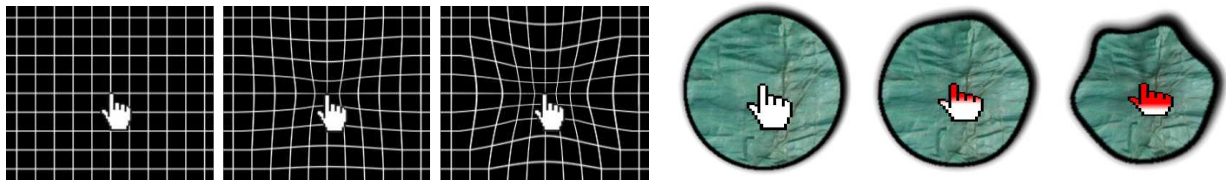


Fig. 1. Elastic Image simulation. Left, animation steps for the proposed image-based deformation as the virtual pressure exerted by the user increases. Right, animation steps of the simulation of an Elastic Image with additional visual feedback.

1. INTRODUCTION

In this work, we aim at further physicalizing the interaction with 2D content through the perceptual simulation of the physical properties of images, precisely we focus on stiffness simulation. The simulation of deformable objects is a well-known area of research in Computer Graphics and Haptic communities [Lin and Otaduy 2008]. While haptic feedback enables the user to perceive the physical properties of computer generated objects (such as stiffness or friction) through force-feedback devices, in this work, we explore whether visual feedback alone is able to generate a sensation of stiffness. Supported by the fact that the visual information on surface displacement has been found to be an important stiffness discrimination cue [Srinivasan et al. 1996; Drewing et al. 2009], we present a novel pseudo-haptic feedback technique based only on visual feedback. The proposed approach enables the simulation of the elastic

properties of images. When interacting with an *Elastic Image*, users will be able to perceive the local elastic properties of the image through a haptic illusion. The elasticity sensation is generated by a procedural image deformation algorithm which modifies the image according to its simulated physical properties and the virtual pressure exerted by the user (see Figure 1). The approach does not require any haptic device, as the simulated pressure depends on the time the user keeps the mouse button pressed. We conducted a psychophysical experiment to quantify our approach and measure its perceptual threshold (or its Just Noticeable Difference). The results showed that users were able to recognize up to eight different stiffness configurations and confirmed that it provides a perceivable and exploitable sensation of elasticity.

We chose a procedural approach due to the fact that the phenomenon we want to simulate is non-linear by nature. Finite element methods would not probably provide real-time simulations (and/or difficult to parameterize) and mass-spring methods would result in a non-uniform behavior. In contrast, the complexity of our approach increases linearly with the number of pixels deformed. This ensures its usage in lightweight devices such as mobile phones or tablet computers and also provides a uniform behavior. Furthermore, in order to reinforce the deformation, we propose additional visual effects such as shadows and animated mouse cursors.

In the remainder of the paper, Section 2 reviews related work on stiffness perception and pseudo-haptics. Section 3 details the Elastic Images approach. Section 4 describes and discusses a psychophysical experiment meant to qualify the perception of local stiffness and Section 5 presents the concluding remarks.

2. PREVIOUS WORK

2.1 Stiffness Perception

Stiffness perception of real and virtual objects has been widely studied in the haptics community. The most common study is to measure the capacity of humans to perceive different levels of stiffness using a dedicated haptic device. As one of the first contributions [Jones and Hunter 1990] demonstrated the capacity of the human haptic system to discriminate changes in stiffness using a force feedback device attached to the arm of the participants. Their results showed that for the force range of 0.67 N/mm to 6.26 N/mm the Weber's fraction for stiffness was stable and its mean value was 0.23. The studies in the literature mainly differ on (1) the muscle groups involved (arm [Jones and Hunter 1990], wrist [Cholewiak et al. 2008], finger [Gurari et al. 2009]), (2) the range of stiffness evaluated (from 0.2 N/mm to 0.5 N/mm [Cholewiak et al. 2008] and [Gurari et al. 2009], up to 0.67 N/mm to 6.26 N/mm [Jones and Hunter 1990]) and (3) the feedback available to the user (visual [Gurari et al. 2009], proprioceptive [Gurari et al. 2009] or only haptic [Cholewiak et al. 2008]). For these studies, there is a lot of discrepancies on the obtained Weber fractions ranging from 0.034 up to 0.99.

A direct application for stiffness simulation is medical training. [Forrest et al. 2009] explored stiffness discrimination for a medical palpation task using a PHANToM device attached to one finger within a "clinically relevant" range of stiffness (from 0.2 to 0.5 N/mm). In their setup, users did not have any visual feedback. Experienced users (veterinarians) were able to perfectly discriminate 2 stiffness levels while non-experienced users (students) were only able to distinguish 1 stiffness level. Other studies have also pointed out that stiffness discrimination is higher for trained users [Cholewiak et al. 2008].

However, stiffness perception can be biased due to additional feedback. For example, as demonstrated by [Srinivasan and LaMotte 1995] when a conflict between haptic and visual feedback appears, there is a visual dominance for stiffness perception. This visual dominance allows the simulation of a wider amount of stiffness levels by the combination of haptic and visual feedback. A similar approach was used by [Moody et al. 2008] for an arthroscopy simulator. In their simulator the hardness of interactive objects was modified through the adaptation of the visual cues presented

to the user. Lécuyer et al. [2001] carried out a user study to quantitatively investigate the effect of visual and haptic dominance on the perception of stiffness. Again, they observed a visual dominance over the haptic stimuli, although large haptic differences could only be compensated by greater visual differences. This result is further supported by [Drewing et al. 2009] where users were able to infer the stiffness of several physical objects only through indirect visual information.

2.2 Pseudo-Haptic Feedback

The definition of Pseudo-haptic feedback as defined by [Lécuyer 2009]: “... *is a technique meant to simulate haptic sensations ... using visual feedback and properties of human visuo-haptic perception*”. Under minimal haptic stimuli, pseudo-haptic feedback can be used in order to enhance haptic sensations (up to some extent). In addition to stiffness [Lécuyer et al. 2001], other studies have proved that pseudo-haptic feedback can be used to simulate other haptic sensations, such as friction or the mass of virtual objects (see [Lécuyer 2009] for a detailed survey).

Pseudo-haptic feedback has also been applied for the perception of relief of 2D images and textures using mouse based input [Lécuyer et al. 2004; Argelaguet et al. 2012], known as Pseudo-Haptic Textures. By modifying the control-display ratio (CD ratio) of the mouse cursor when exploring a 2D image, it becomes possible to simulate negative or positive slopes and allowing the user to “perceive the relief” of the image. An orthogonal approach proposed by [Lécuyer et al. 2008] to reinforce the relief of the image was based on modifying the size of the mouse cursor as a function of the depth encoded in the image. Additional work has focused in the addition of visual vibrations to the cursor, enhancing the perception of the relief for textures with stripped patterns [Hachisu et al. 2011].

2.3 Physicizing User Interfaces

Current graphical user interfaces (GUIs) are driven by mouse, pen or touch-based inputs (we are not considering key input). However, in general, the input provided by these devices does not provide any relevant haptic sensation (feedback) to the user. Work relevant to ours are these which enhance the expressiveness and the perception of GUIs. For example, specialized touch devices which provide haptic feedback such as programmable friction surfaces [Levesque et al. 2012] and tactile surfaces [Jansen et al. 2011] which increase information transfer between the system and the user. The different properties perceived by the user can be mapped to a specific meaning [Chan et al. 2008].

In contrast, pressure-based input allows the information transfer in the opposite direction. The monitor of the pressure exerted by the user through pressure sensors in a mouse device [Cechanowicz et al. 2007] or through pen-based interfaces [Ramos et al. 2004] allows the user to trigger different states according to the pressure exerted. Mizobuchi et al. [2005] showed that users were able to generate reliably six different pressure levels using pen-based interfaces (corroborating the results of [Cechanowicz et al. 2007; Ramos et al. 2004]). In addition, they also determined that the range of optimal pressure exerted by users is between 0 and 3 N.

The main challenge of this work is to create a pseudo-haptic effect of stiffness when interacting with 2D content without the need of dedicated hardware. The sticky widgets approach of Rodgers et al. [2006] showed the potential of applying a pseudo-haptic approach for the interaction with GUIs. In their work, by dynamically adjusting the CD ratio of the mouse cursor, widgets that could be potentially acquired by the user presented a sticky behavior in order to improve its acquisition.

3. ELASTIC IMAGES

In this section we present our novel pseudo-haptic approach to modify the perceived local stiffness of a 2D image, thereafter referred to as Elastic Image. When the user clicks on an Elastic Image, the area surrounding the mouse cursor is deformed simulating that the user is pushing the image. Furthermore, we propose additional effects to reinforce the pseudo-haptic effect, such as procedural shadows, non-uniform deformations and the animation of the mouse cursor.

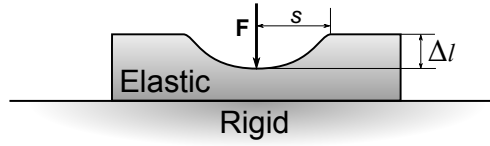


Fig. 2. Considered deformation model. When applying a force F to an elastic material, it will sink a length Δl and the deformed area will have a spread of radius s .

3.1 Concept

An elastic object presents a deformation when we apply a force on it (see Figure 2). The deformation depends on the physical properties of the object such as its stiffness and compressibility. The deformation is characterized by the force F required to deform the material, a length Δl , and the spread of the deformation s . The relationship between the force and the length is determined by the stiffness coefficient (k) through *Hooke's law* : $F = k \cdot \Delta l$. In our scenario, an Elastic Image has predefined physical properties which will drive an image-based deformation.

In the first place, we have to determine the force exerted by the user. As we are not using any haptic or pressure input device, we propose the virtual pressure exerted by the user to be proportional to the press time. For a mouse based interface, the force applied will be proportional to the time the user holds the mouse button pressed. Second, we must provide a plausible and a smooth image-based deformation simulating the effect of pushing the Elastic Image towards the screen. Finally, it must support the deformation of images with different elastic properties. The user has to be able to perceive whether two Elastic Images have different properties. All the simulated deformations stay in the elastic phase, when the user releases the mouse button, the deformation is reversed and the image will recover its original state.

3.2 Texture Deformation Algorithm

The effect of pressing an image towards the screen is achieved by displacing the pixels towards the mouse cursor. Once the user presses the image, a smooth animation at pixel level is generated. We first detail the image-based algorithm and later on we discuss how to modify it in order to generate different material properties.

The deformation algorithm computes for each pixel of the image $p = (r, \theta)$ its new color $p_c((r, \theta), t)$ at the time t of the deformation animation (see Equation 1) (when $t = 0$ it matches the original image). We use a polar coordinate system where the origin is determined by the mouse cursor. In Equation 1, \mathbf{v} is the normalized vector defined by the mouse cursor position $m = (0, 0)$ and the pixel position (r, θ) and the pixel displacement $d(r, t)$.

$$p_c((r, \theta), t) = p_c(\mathbf{v} \cdot d(r, t) + (r, \theta), 0) \quad (1)$$

The amount of the displacement for each pixel $d(r, t)$ (see Equation 2) depends on the distance r between p and m (normalized) and the animation time $t \in [0..1]$. The animation starts at $t = 0$ and ends at $t = 1$. Later on, we discuss how we normalize r and t according to the stiffness coefficient k . λ ranges from 0 to 1.

$$d(r, t) = d_a(r, t)\lambda + d_b(r, t)(1 - \lambda) \quad d(r, t) \in [0..1] \quad (2)$$

The final deformation is obtained through the combination of two deformations (see Equations 3 and 4). The deformation defined by d_a exhibits a quadratic profile when $r < t/2$ and a linear profile when $r \geq t/2$ (see Figure 3 left). We explored the usage of different profiles (linear, gaussian and quadratic) being this combined approach the

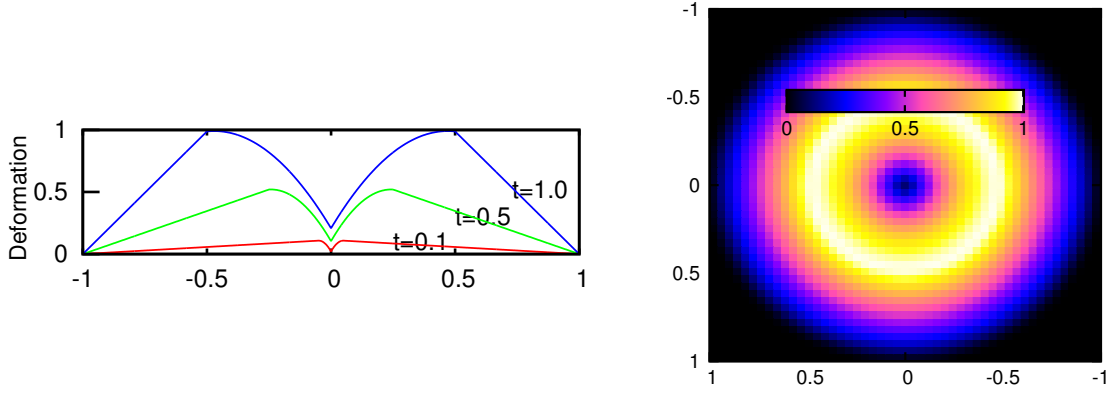


Fig. 3. Left, section of the 3D plot of Equation 2 for different values of t . Right, plot of Equation 2 when $t = 1$. It represents the amount of image displacement (normalized) for each pixel in the deformed area. The point (0,0) represents the mouse location.

one which minimized texture artifacts and provided a more realistic behavior. However, although this deformation provides a smooth result, the pixels closer to the mouse cursor exhibited low displacements. In order to provide a subtle but noticeable translation of these pixels, we introduced $d_b(r, t)$. This function adds a constant translation over time, reinforcing the deformation effect. According to preliminary tests, a value of $\lambda = 0.8$ for Equation 2 provided the desired deformation.

$$d_a(r, t) = \begin{cases} 0 & \text{if } t = 0 \\ \left[1 - \left(\frac{2r}{t} - 1 \right)^2 \right] t & \text{if } r < t/2 \\ \frac{r - 1}{0.5 - 1/t} & \text{if } r \geq t/2 \end{cases} \quad (3)$$

$$d_b(r, t) = t \cdot (1 - r) \quad (4)$$

3.3 Stiffness Model

In order to simulate different levels of stiffness, the deformation must be adapted accordingly. Our deformation algorithm considers three different parameters: (a) the spread of the deformation, (b) the maximum pixel displacement and (c) the force required to reach the maximum deformation. These parameters can be modified to simulate different levels of stiffness. For example, a soft material will have a bigger deformation area, it will have a greater pixel displacement and will require less force to deform.

Equation 2 requires two normalized input variables r and t . The normalization of these variables determines the area of influence of the deformation (r) and the animation time (t). In addition, the value of $d(r, t)$ is also normalized and has to be transformed into pixel displacement.

According to our deformation scheme, we performed several informal tests to determine the minimum and maximum values of the three parameters. For the deformation strength, we considered the perceptible amount of deformation that the user can perceive and the limit of the deformation in terms of texture artifacts, determining a suitable range in between $d_{min} = 10\text{px}$ and $d_{max} = 30\text{px}$ (the test was performed in a monitor with squared 0.27mm pixel

size). Regarding the area of influence, the deformation should be visible but keeping the deformation locally, the range considered was $r_{min} = 50\text{px}$ to $r_{max} = 90\text{px}$. Finally, for the animation time, the time should be long enough to allow the user to perceive the animation but not too long, the animation time ranged from $t_{min} = 0.25\text{s}$ to $t_{max} = 2\text{s}$. All three variables vary linearly according to the stiffness coefficient of the image ($k \in [0..1]$) which determines the behavior of the animation. Our assumptions are that the animation time is proportional to the stiffness, stiffer materials reach their elastic limit slower than less stiff materials (time is related to the force). On the other hand, the deformation strength and the area of influence are inversely proportional to the stiffness. Although the area of influence depends on the compressibility properties of the material simulated, we assume that both are correlated. Furthermore, as we want to keep the deformations in the elastic phase, when $t = 1$ the deformation stops. We can assume that after $t = 1$ a non-deformable object placed underneath the object avoids further deformations.

Finally, once the user releases the mouse button, the animation is reversed, but the time required to reach $t = 0$ is modified. The material property that models the relaxation is the viscosity. If needed, the viscosity coefficient could be also determined independently for each image. In our experiments we assumed that the viscosity is inversely proportional to the stiffness, thus stiffer materials will require less time to recover than less stiff materials. We considered relaxation times to be smaller than pressing times, for the experiments during the release phase the animation time ranged from $t_{min} = 0.125\text{s}$ to $t_{max} = 0.5\text{s}$.

3.4 Shadows and Creases

We provide two extensions of the deformation algorithm in order to increase the realism of the deformation. First, we propose a method to generate shadows by modifying the luminance of the area being deformed, in order to reinforce the effect that the texture is being pushed towards the screen. And second, we detail a method to simulate the deformation of materials having a non-uniform deformation.

The lighting conditions when pressing a surface vary locally. Light intensity vary due to local occlusions (ambient occlusion) and due to the change of the normals of the surface. Similar effects are applied in 2D GUIs elements to simulate relief. However, as we do not have any a priori information about the shape of the surface or the origin of the light sources we only consider changes on the ambient occlusion in order to compute luminance changes. In our approach, the illumination changes according to t and r describing a Gaussian curve (see Equation 5). The decrease in luminance is more visible at the center of the deformation and as the deformation advances. The effect will increase the depth perception of the user and reinforces the effect of pushing the image.

$$l(t, r) = 1 - \frac{e^{(r \cdot \sqrt{t} - 1)/2}}{(2 \cdot 0.12)^2} \quad (5)$$

The second strategy is based on the fact that the deformation of the material might not be perfect: only pure elastic materials will deform uniformly. Typically, creases are generated on the surface according to the compressibility of the material (e.g. when pressing a piece of fabric). Our goal was to generate a non-uniform deformation and update the generated shadow according to the properties of the surface. The effect is achieved by adding a new modifier $c(\theta) \cdot r$ to the maximum allowed deformation of Equation 2 as defined in Equation 6.

$$d(r, \theta, t) = [d_a(r, t) \cdot c(\theta) \cdot r] \cdot \lambda + d_b(r, t) \cdot (1 - \lambda) \quad (6)$$

Several $c(\theta)$ functions can be defined in order to achieve different effects. We used the combination of several cosine functions in order to generate the new deformation profile (see Figure 4 left and Equations 7 and 8). In addition, a random offset ($[0..2\pi]$) can be added to θ to modify the creases' orientation, for example, after each button press.

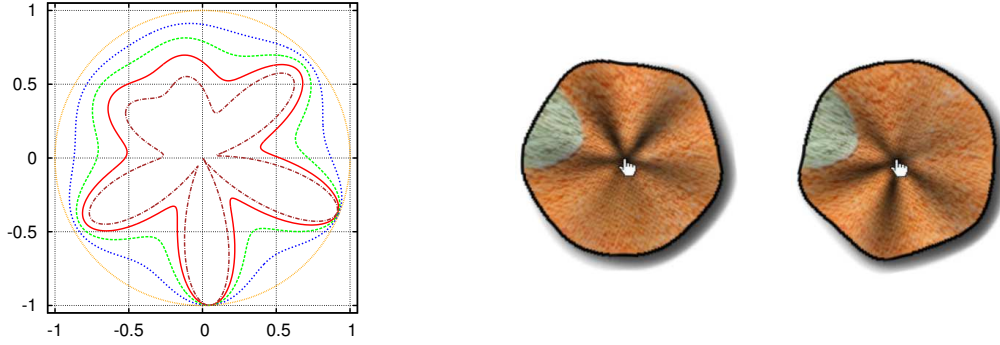


Fig. 4. Procedural creases. (Left) Plot of the sinusoidal functions used to simulate the creases defined in Equations 7 and 8 with $k = \{2, 4, 8, 20\}$. The same function with different weights can be used to simulate different materials. (Right) Two examples of the procedural creases adding a different randomized offset to θ .

$$c(\theta) = ||(\cos(\theta) f(\theta), \sin(\theta) f(\theta))|| \quad c(\theta) \in [0..1] \quad (7)$$

$$f(\theta) = \left\{ \frac{\cos(6\theta + 0.5) - \cos(5\theta) + k}{k + 2} \right\} \quad (8)$$

As an indirect effect, the addition of the creases required another approach to generate the shadows (see Figure 4 Right), the creases also influence in how the ambient occlusion is computed. Now, the shadow is generated by modifying the luminance according to $c(\theta)$ (see Equation 9).

$$l(t, \theta, r) = c(\theta) \cdot t \cdot (1 - r) \quad (9)$$

3.5 Handling Multiple Objects

The description of the algorithm so far, only considers the deformation of a texture representing one material (the texture has a given stiffness coefficient). However, consider a photograph (or a complex GUI) composed of several objects with different stiffness coefficients each. In order to simulate the different levels of stiffness, we have to encode the stiffness of each object and we must modify the deformation algorithm to take into account their boundaries.

Our first approach was to limit the deformation at the boundaries of the objects using a distance map encoding the distance from each pixel to the boundary of the object. However, informal user's testing showed that users felt that the behavior was non-realistic, as the boundaries did not deform properly. Instead, we propose an alternative solution which provides more flexibility, better handles the deformation at the boundaries and can be directly applied in GUIs. We propose to split the different objects in the scene into separate layers in order to handle their deformation independently. The deformation of an object is constrained to the active layer and each layer has its own stiffness coefficient. Although layers might overlap, we restrict the interaction to visible areas and these considered interactive. By default only non-transparent pixels are considered interactive. For example, in Figure 5 when the user clicks on the shadow areas, no deformation will be generated. However, this behavior can be overridden by providing an additional mask for each layer. For example, if we are simulating a transparent object we will need to define a layer mask with its boundaries. When the user clicks a pixel in a non-interactive area of a layer, the system will omit that layer and will process the layer below. Furthermore, if needed, the deformation of a layer can be propagated into the layers below (e.g. the layer simulates a thin material such as a piece of fabric).

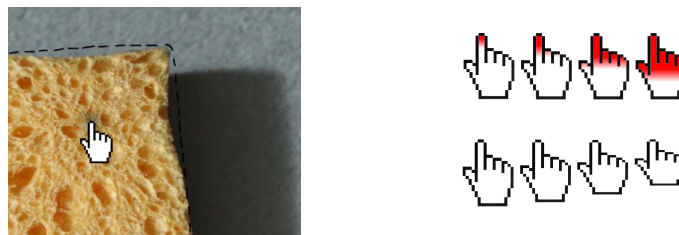


Fig. 5. Elastic Images extensions. (Left) Close up of the deformation of the boundaries for an Elastic Image. The background image (in a different layer) remains unmodified. (Right) Visual feedback alternatives for the mouse cursor while pressing an Elastic Image. The pressure increases from left to right. (Right top) Effort indication. (Right bottom) Gesture indication.

3.6 Mouse Cursor Animation

The standard mouse cursor is a graphical element which does not provide any additional visual feedback while the user is clicking. When interacting with an Elastic Image, we propose to increase the visual feedback of the mouse pointer reinforcing the action of pressing an image. The goal is to provide a force indicator and a gesture indicator through the animation of the mouse cursor. In our deformation scheme, as the simulated force is proportional to the time the user keeps the mouse button pressed, we propose to animate the mouse cursor considering the press time. The mouse cursor will vary over time displaying the force applied until the deformation finishes. Stiffer materials will have longer animation times than softer materials. We designed an animated cursor of a pointing hand, which can be seen as a metaphor for a touching gesture. Two different alternatives were considered. First, modifying the color of the cursor from the tip of the finger, generating an impression of progressive effort (see Figure 5 right top). An additional benefit of this approach is that it resembles to a progress bar. Second, modifying its shape to simulate the effect of pushing the image (see Figure 5 right bottom). Both alternatives can be combined.

3.7 Summary

In this section, we have detailed the visual feedback and the implementation details of the Elastic Images approach. We have proposed a lightweight method to compute a plausible elastic deformation of an image, and several additional visual feedback effects to reinforce the effect of “pushing” a Elastic Image. In our implementation, the deformation algorithm runs entirely on the GPU (using a fragment shader) and the overall performance overhead is negligible. The proposed approach can be easily implemented in any device with OpenGL ES 2.0 support. In the accompanying materials, we provide an interactive demonstrator¹ of the Elastic Images and a video depicting the approach.

4. USER EVALUATION

We conducted a user evaluation to explore whether the Elastic Images approach provides enough exploitable and perceptible information about the simulated physical properties. The evaluation is not considering whether the deformation is realistic, appealing or how it influences user performance [García et al. 2010]. The evaluation focused on stiffness discrimination, exploring if users were able to discriminate whether two Elastic Images had the same stiffness (using the same texture). This will provide us information about the viability of the approach, and explore if the Elastic Images are able to create a perceptual illusion of stiffness.

As detailed in Section 3.3, the deformation algorithm is driven by three parameters, the deformation strength (D), the deformation radius (R) and the animation time (T). We explored three different configurations, each increasing the amount of feedback provided. We explored the user’s perception when modifying only the deformation strength (D), when modifying the deformation strength and the radius at the same time (DR) and all three parameters (DRT).

¹<http://team.inria.fr/hybrid/elastic-images>



Fig. 6. Example for the deformation of the sponge texture used during the experiments.

The results will provide us with information about the amount of variation of the stiffness coefficient required to have a perceptible difference (Just Noticeable Differences), and also whether the visual feedback provided unequivocally allows to identify the stiffness of an Elastic Image. In this experiment we only considered the basic image deformation algorithm. We did not include the additional effects described such as the creases, the shadows or the animated mouse cursor. We believe that the critical aspect of the approach is the texture deformation algorithm, the other effects will potentially reinforce the effect. However, it will be interesting in the future to be able to quantify the reinforcement of the stiffness perception.

4.1 Procedure

The task carried out by the participants was a classical 2AFC (2 alternative forced choice) task from psychophysics. For each trial, two Elastic Images (*reference and comparison*) were presented to the participants. Each Elastic Image had the same texture (see Figure 6) but different stiffness coefficients. Both images were displayed at the center of the screen and only one image was shown at once. The procedure had five different steps: (a) the first alternative is presented; participants press the Elastic Image with the mouse until the image is fully deformed. (b) Participants release the mouse button and wait until the image returns to its original state. (c) The second alternative is presented; participants press the Elastic Image again. (d) Participants release the mouse button and wait until the image is restored. (e) Finally, participants choose which image was the stiffest (two buttons are provided for this purpose). If the participant was not sure about which was the stiffest, they were instructed to choose an answer randomly. In order to ensure that all users experienced the same deformation, they were forced to press them at the same screen coordinates and they have to press the target until it fully deforms. Participants were never told about what was supposed to be a stiffer image, participants had to infer this information only through the visual feedback provided.

4.2 Design and Hypothesis

The design of the experiment followed a JND (just noticeable differences) methodology [Gescheider 1985]. The JND experiment requires to choose several reference values for the measured stimuli (*reference targets*) and compare them with a fixed window around them (*comparison targets*). The stimuli considered was the stiffness coefficient (k).

The values of the stiffness coefficient for the *reference targets* were $k_r = \{1/3, 2/3\}$. We chose them in order to explore whether the behavior is symmetric: are stiffer objects easier to discriminate than softer ones? Regarding the *comparison targets*, we defined six for each *reference target*, $\Delta k = \{-0.24, -0.16, -0.08, 0.08, 0.16, 0.24\}$. The stiffness coefficient for each trial k_t was defined as $k_t = k_r + \Delta k$, resulting in twelve different combinations. For conditions D and DR, then the area of influence (R) and the animation time (T) did not depend on the stiffness coefficient, the radius of the deformation was computed as $(r_{min} + r_{max})/2$ and the total animation time as $(t_{min} + t_{max})/2$.

According to the factors described, the independent variables considered for the study were: (1) the computation of the deformation parameters (*Technique*: D, DR and DRT), (2) the stiffness coefficient for the reference targets (*Reference*: $k_{r[1|2]}$) and (3) the difference between the *reference* and the *comparison target* (*Comparison*: $\Delta k_{[1|2|3|4|5|6]}$). All

variables were within-subjects. In total, this resulted in three factors with 3, 2 and 6 levels respectively. We performed 20 repetitions of each combination, resulting in a total of 720 trials. To avoid bias towards the first or the second target, users might choose always the same target in case they do not know which object is stiffer, we balanced the order of appearance of the reference target. For each combination, it appears first for half of the trials and vice-versa. The order of the trials was also randomized.

The dependent variables were (1) the accuracy and (2) the response time. The accuracy is the rate of right answers (the user chooses the stiffest object) while the response time is the time passed from the release of the mouse button after pressing the second target until the user chooses an answer. The following hypotheses were considered:

H1 Participants will have better accuracy rate for DR technique in comparison with D.

H2 Participants will have the best accuracy rate for DRT technique.

H3 Participants will obtain the same accuracy for each reference.

Participants needed around one hour and a half to finish the experiment. Due to the length of the experiment and the concentration requirement they were asked to take all the breaks they wanted. Also at the beginning of the experiment a short training session was conducted to ensure that they understand the procedure.

4.3 Apparatus and Participants

The experiments were conducted in a 17 inch monitor with a resolution of 1280×1024 pixels, using as an input device a standard mouse. The distance between the user and the monitor was $40cm$. Nine participants aged from 22 to 31 ($\bar{x} = 25.88$, $SD = 2.977$), eight males, participated in the experiment. All participants had no previous experience.

4.4 Results

First, we will discuss the results of the ANOVA analysis for the time and accuracy measurements. For all post-hoc comparisons, Bonferroni adjustments for $\alpha = 95\%$ were applied. Only significant post-hoc comparisons are mentioned ($p < 0.05$).

Accuracy. The accuracy value for each combination of factors was computed by considering the number of right choices divided by the number of repetitions (see Figure 7 left). The three-way ANOVA showed a main effect for Technique ($p < 0.001$; $F_{2,8} = 65.34$) and Comparison ($p < 0.001$; $F_{5,8} = 41.00$). No significant effect was found for Reference ($p = 0.44$), thus we can not reject H3.

Post-hoc tests revealed that users were significantly more accurate when simulating the deformation with the DRT method, thus accepting H2. However, no significant differences were found between D and DR, thus rejecting H1. For the Comparison factor, the post-hoc tests showed that the accuracy increased significantly as the difference in stiffness (absolute value) between the *reference* and the *comparison* image increases. In contrast, when the difference (absolute value) is zero, no significant differences were found (± 0.24 , ± 0.16 and ± 0.08). The ANOVA also showed a two-way interaction effect between Technique and Reference ($p < 0.05$; $F_{2,8} = 4.26$). Although, post-hoc tests did not show any significant difference, for the DRT approach, we observed that in overall, users had higher accuracy rates for softer images ($k_r = 1/3$, $\bar{x} = 0.87$) than for stiffer ones ($k_r = 2/3$, $\bar{x} = 0.82$).

Time. Regarding the reaction time, the three-way ANOVA only showed one main effect for Technique ($p < 0.05$; $F_{2,8} = 6.1$). The mean reaction times were $1.65s$ for D, $1.72s$ for DR and $1.75s$ for DRT. Post-hoc tests only showed a significant difference between the techniques D and DRT.

JND. The second step is to analyze which is the minimum noticeable difference that can be perceived by users. The JND results will give us an insight about the minimum difference of the stiffness coefficient that can be efficiently

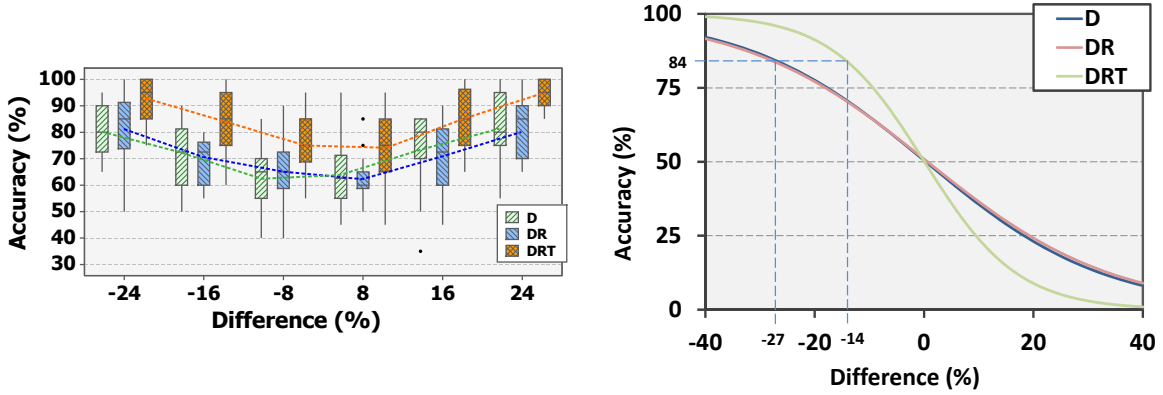


Fig. 7. (Left) Boxplot of accuracy grouped by *Technique* and *Comparison*; dashed lines connect mean values. The boxplot shows that accuracy increases as differences in stiffness increases and that accuracy rates for the DRT technique were the highest. (Right) Plot of the psychometric curve for each Technique. The PSE for all curves matches the condition where the difference between the reference and the comparison is zero.

discriminated. Now, instead of considering the accuracy as is, we consider the amount of times that the reference target was considered to be stiffer. As the ANOVA study did not found any difference between references, we did not split the data for each reference and only consider the Technique as a factor.

The Weber fraction for each technique was (D) 0.27, (DR) 0.29 and (DRT) 0.14 (the threshold for computing the Weber fraction was 84%). The Weber fraction was obtained fitting the psychometric curve $f(x) = 1/(1 + e^{(x-\alpha)/\beta})$ to the data grouped by Technique (see Figure 7 Right). The α and β values for each factor were: (D) $\alpha = 0.3$ and $\beta = 16.36$, (DR) $\alpha = 0.613$ and $\beta = 16.98$, (DRT) $\alpha = 0.071$ and $\beta = 8.56$. Similar to the ANOVA, which only showed better accuracy rating for the DRT condition, the Weber fraction shows that users are able to detect changes in the stiffness coefficient k easily for the DRT technique.

5. DISCUSSION

The main result of this experiment is that users, although they were not trained regarding what was supposed to be a stiffer image, were able to infer the stiffness of the image only through the visual feedback provided. These results show that we were able to generate the pseudo-haptic effect of stiffness on the image.

Regarding the different techniques evaluated, we can state that the DRT approach is the one which allows to better distinguish smaller changes on the stiffness coefficient. Our believe is that the change of the deformation time provides an improved recognition on itself. However, using only the animation time to customize the deformation might be confusing for the user as all images will deform in the same way no matter their stiffness. This is the reason why we did not consider this approach in the experiment.

Although the ANOVA did not show any main effect for the Reference factor, it seemed that users had less trouble on differentiating softer images that stiffer ones. As the feedback provided was only visual, we did a preliminary analysis on the visual flow generated during the deformation. From the analysis we made two observations, first, the visual flow for a given stiffness coefficient remains constant during the entire animation. Second, when varying the stiffness coefficient, the visual flow does not vary linearly with respect to the stiffness coefficient. As we vary the stiffness coefficient, the changes on visual flow are higher for smaller values of stiffness. This is a possible explanation about why we observed that users have less difficulties for softer images, although additional analyses are required.

The recognition time results did not provide conclusive results. The differences in recognition time were small (around 0.1 seconds). Furthermore, the absence of a main effect on *Comparison* does not allow to correlate the recognition time with the users' confidence. Users answered fast, no matter their confidence.

Finally, taking into account that the stiffness coefficient is normalized and the Weber fraction of the stiffness coefficient for the DRT method was 0.14, we would be able to simulate approximately 8 different stiffness levels ($k = 0, 0.14, 0.28, \dots, 1$) which will be perceivable for users.

5.1 Subjective Questionnaires

At the end of the experiment we asked participants to fill a questionnaire to gather additional information. Regarding the deformation algorithm, most of the participants stated that they were able to perceive the differences between all three parameters (deformation, radius and duration), but all agreed that changes for the DRT condition were easier to perceive. We also asked which was their strategy used during the evaluation, most of them reported that they focused on how the image was deformed over time rather than the final state of the deformation. In addition, some participants also reported that they perceived that the image was pushed towards the screen (*"It was like pushing the sponge with the finger"*). In summary, the subjective data gathered in the questionnaires supports the fact that the accuracy rates for the DRT condition were significantly higher and show that the deformation algorithm is able to simulate the effect of "pushing" an image with varying stiffness coefficients.

6. CONCLUSION

In this paper we have proposed and evaluated a pseudo-haptic approach for the simulation of the local elasticity of images. The stiffness simulation is generated by an image-based deformation algorithm which is able to efficiently provide a haptic illusion of stiffness. The deformation is generated procedurally at pixel level using only original image as an input. The user, in order to perceive the local stiffness of an Elastic Image, only requires the mouse as an input device. Once the user clicks on the image, the deformation is driven by the stiffness coefficient of the image and the time the user keeps the mouse button press (the force is proportional to the button press). In addition to the basic deformation algorithm, we have also proposed additional effects to reinforce the effect such as the use of shadows or the animation of the mouse cursor. The results from the user evaluation showed that users are able to efficiently distinguish variations of the stiffness coefficient up to 14%. Considering the range of potential stiffness coefficients (from 0 to 1), our approach is able to simulate up to eight different stiffness coefficients. Users were able to determine which image was stiffer, not only that there was a difference in the deformation.

Regarding its applications, we believe that the use of the Elastic Images can be further extended not only focusing on perceptual simulation. Graphical user interface elements, such as buttons, can be configured with different stiffness properties. The interface designer, by providing different levels of stiffness to the elements of the graphical user interface, can associate the stiffness to different behaviors. For example, stiffer elements can be provided for actions that cannot be undone or potentially harmful to the system.

In order to extend the current work, several directions can be pursued. First, additional user studies have to be conducted to explore how the additional visual feedback effects proposed reinforce the stiffness perception. In addition, we must explore how the underlying image biases the stiffness perception. If two Elastic Images have the same stiffness coefficient but a different texture, which is the dominant visual stimuli? Furthermore, although we have only taken into account the use of a mouse as an input device, it will be interesting to evaluate how the Elastic Images behaves for tactile and pressure based interfaces. It remains unclear how the couple between the Elastic Image and the user's finger will modify the user's perception.

REFERENCES

- ARGELAGUET, F., GÓMEZ JÁUREGUI, D. A., MARCHAL, M., AND LÉCUYER, A. 2012. A Novel Approach for Pseudo-haptic Textures Based on Curvature Information. In *Eurohaptics'12. Haptics: Perception, Devices, Mobility, and Communication*. 1–12.
- CECHANOWICZ, J., IRANI, P., AND SUBRAMANIAN, S. 2007. Augmenting the mouse with pressure sensitive input. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. CHI '07. 1385–1394.
- CHAN, A., MACLEAN, K., AND MCGRENERE, J. 2008. Designing haptic icons to support collaborative turn-taking. *International Journal of Human-Computer Studies* 66, 5, 333–355.
- CHOLEWIAK, S. A., TAN, H. Z., AND EBERT, D. S. 2008. Haptic identification of stiffness and force magnitude. In *Proceedings of the Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. 87–91.
- DREWING, K., RAMISCH, A., AND BAYER, F. 2009. Haptic, visual and visuo-haptic softness judgments for objects with deformable surfaces. In *Proceedings of the 2009 World Haptics Conference (WHC09)*. IEEE, 640–645.
- FORREST, N., BAILLIE, S., AND TAN, H. 2009. Haptic stiffness identification by veterinarians and novices: A comparison. In *Proceedings of the 2009 World Haptics Conference (WHC09)*. IEEE, 646–651.
- GARCÍA, M., OTADUY, M. A., AND O'SULLIVAN, C. 2010. Perceptually Validated Global/Local Deformations. *Computer Animation and Virtual Worlds* 21, 3-4, 245–254.
- GESCHIEDER, G. A. 1985. *Psychophysics: Method, Theory, and Application*. Lawrence Erlbaum Associates, New Jersey, US.
- GURARI, N., KUCHENBECKER, K. J., AND OKAMURA, A. M. 2009. Stiffness discrimination with visual and proprioceptive cues. In *Proceedings of the 2009 World Haptics Conference (WHC09)*. IEEE, 121–126.
- HACHISU, T., CIRIO, G., MARCHAL, M., LÉCUYER, A., AND KAJIMOTO, H. 2011. Pseudo-haptic feedback augmented with visual and tactile vibrations. In *Proceedings of the International Symposium on VR Innovations (ISVRI'11)*. 331–332.
- JANSEN, Y., KARRER, T., AND BORCHERS, J. 2011. Mudpad: Tactile feedback for touch surfaces. In *CHI EA '11: Extended Abstracts on Human Factors in Computing Systems*. ACM, 323–328.
- JONES, L. A. AND HUNTER, I. W. 1990. A perceptual analysis of stiffness. *Experimental Brain Research*, 150–156.
- LÉCUYER, A. 2009. Simulating haptic feedback using vision: A survey of research and applications of pseudo-haptic feedback. In *Presence: Teleoperators and Virtual Environments*. Vol. 18. MIT Press, 39–53.
- LÉCUYER, A., BURKHARDT, J. M., COQUILLART, S., AND P., C. 2001. Boundary of illusion: An experiment of sensory integration with a pseudo-haptic system. In *Proceedings of the IEEE International Conference on Virtual Reality*. 115–122.
- LÉCUYER, A., BURKHARDT, J.-M., AND ETIENNE, L. 2004. Feeling bumps and holes without a haptic interface: the perception of pseudo-haptic textures. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. CHI '04. ACM, 239–246.
- LÉCUYER, A., BURKHARDT, J.-M., AND TAN, C.-H. 2008. A study of the modification of the speed and size of the cursor for simulating pseudo-haptic bumps and holes. *ACM Transactions on Applied Perception* 5, 3, 14:1–14:21.
- LEVESQUE, V., ORAM, L., AND MACLEAN, K. 2012. Exploring the design space of programmable friction for scrolling interactions. In *Proceedings of the IEEE Haptics Symposium 2012*. 23–30.
- LIN, M. C. AND OTADUY, M. A., Eds. 2008. *Haptic Rendering: Foundations, Algorithms, and Applications*. AK Peters.
- MIZOBUCHI, S., TERASAKI, S., KESKI-JASKARI, T., NOUSIAINEN, J., RYNNANEN, M., AND SILFVERBERG, M. 2005. Making an impression: Force-Controlled Pen Input for Handheld Devices. In *CHI EA '05: Extended Abstracts on Human Factors in Computing Systems*. ACM, 1661–1664.
- MOODY, L., WATERWORTH, A., ARTHUR, J. G., MCCARTHY, A. D., HARLEY, P. J., AND SMALLWOOD, R. H. 2008. Beyond the visuals: tactile augmentation and sensory enhancement in an arthroscopy simulator. *Virtual Reality* 13, 1, 59–68.
- RAMOS, G., BOULOS, M., AND BALAKRISHNAN, R. 2004. Pressure widgets. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. CHI '04. 487–494.
- RODGERS, M. E., MANDRYK, R. L., AND INKPEN, K. M. 2006. Smart sticky widgets: Pseudo-haptic enhancements for multi-monitor displays. In *Proceedings of Smart Graphics'06*. 194 – 205.
- SRINIVASAN, M. AND LAMOTTE, R. 1995. The impact of visual information on the haptic perception of stiffness in virtual environments. *J. Neurophysiology* 33, 1, 88–6101.
- SRINIVASAN, M. A., BEAUREGARD, G. L., AND BROCK, D. L. 1996. The impact of visual information on the haptic perception of stiffness in virtual environments. In *Proceedings of the ASME Dynamic Systems and Control Division*. 555–559.